Ananda Cahyo W/07111940000128
SPC – DTE ITS

Tugas 1: Membership Function

Lecturer: Ari Santoso, DEA	Nama: Ananda Cahyo Wibowo
Course: EE184922 – Intelligent Control System	NRP: 07111940000128

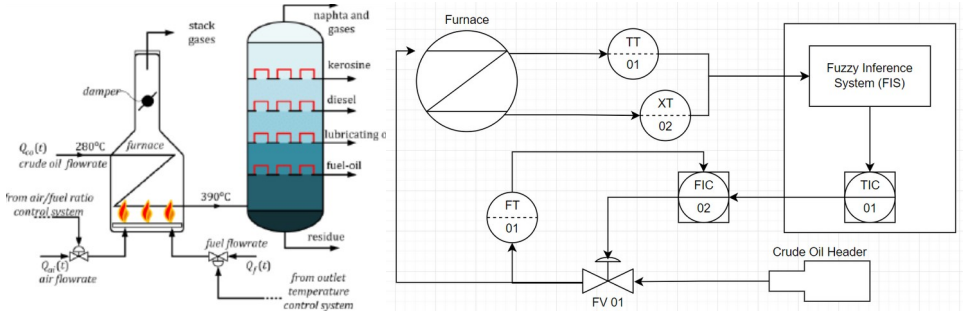
Penugasan

Buatlah Fuzzy Inference System untuk 5 jenis aplikasi yang berbeda. Gunakan minimal 3 kategori linguistic untuk setiap FIS. Gunakan jenis Gaussian (NRP Genap) atau jenis Triangle (NRP Ganjil) untuk jenis Membership Functionnya.

Jawaban

1. Kontrol Kaskade pada Temperatur plant Furnace
   a. Deskripsi

      Furnace atau tungku pembakaran dalam proses industri merupakan komponen yang penting terlebih dalam mengubah fasa zat atau energi. Contohnya adalah proses pembakaran crude oil sebelum masuk kolom destilasi untuk menjadi produk jadi. Jika hanya menggunakan kendali PID konvensional, factor lain akan sulit dimasukan untuk mendapatkan hasil yang sesuai. Oleh karenanya digunakan kendali kompensasi Fuzzy untuk mengoptimalkan proses kontrol.



      Kontrol Cascade memiliki 2 jenis controller yang biasa disebut master dan slave. Pada contoh kasus ini digunakan jenis kendali cascade Fuzzy Temp-Expansion(TIC01) dengan flow controller(FIC02), dimana flow controller dikendalikan oleh controller cascade fuzzy sebagai masternya.

      Nantinya output TIC adalah flow yang harus dikeluarkan control valve yang dalam linguistic fuzzy dibagi low, medium, dan fast.

   b. Membership Function

Input (2 parameter):

- Temperature Furnace (3 Kategori):
  - Warm = 0 – 500 degC
  - Hot = 200 – 1000 degC
  - Too Hot = >900 degC
- Thermal Expansion (3 Kategori):
  - under-expansion = -10 – -5 cm
  - normal = -7 – 7 cm
  - over expansion = >5 cm

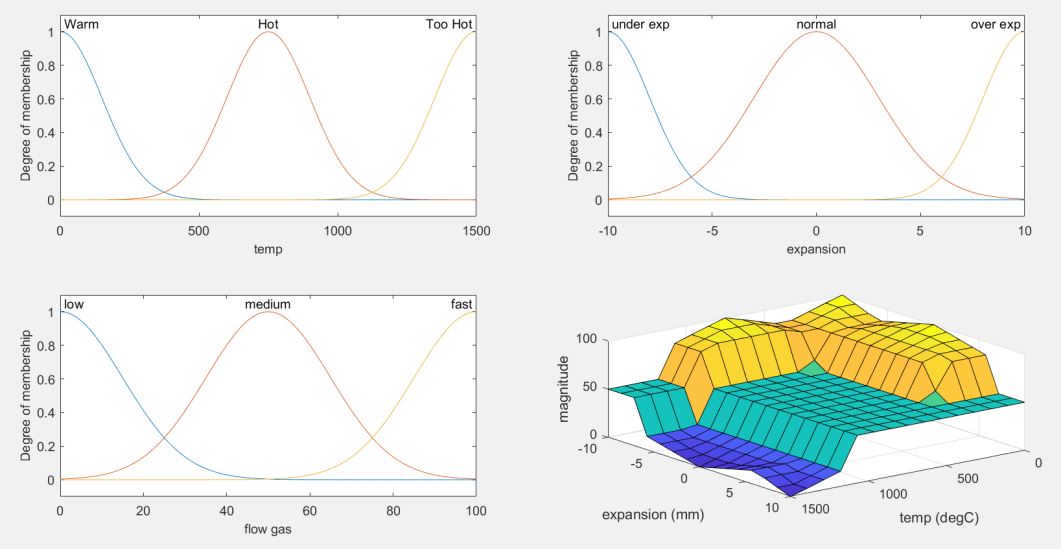
Output(1 parameter):

- Crude Oil Flow (3 Kategori):
  - Slow = 0 – 50 L/H
  - Medium = 30 – 80 L/H
  - Fast = >70 L/H

c. Rule Base

Rule based/aturan dari FIS dapat disusun dalam table sebagai berikut:

T\δ	Under-expansion	Normal expansion	Over-expansion
Warm	Fast	Fast	Medium
Hot	Medium	Medium	Medium
Too Hot	Medium	Slow	Slow

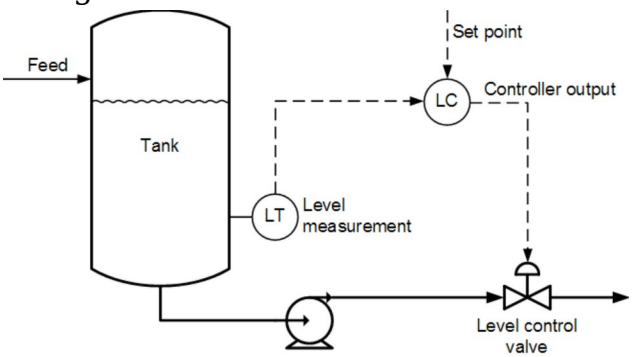


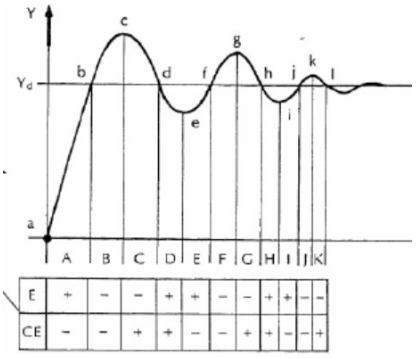
2. Kontrol Feedback pada Opening Valve

a. Deskripsi

Pada pengaturan level menggunakan opening control valve sebagai aktuatornya, dijumpai siste orde 2 yang memiliki respon seperti dibawah. Input dari FIS dapat

diambil dari error dan delta error dari sistem. Opening valve diharapkan dapat mengatur flow air sehingga bisa mengontrol level fluida di dalam tank.





b. Membership Function

Input (2 Parameter):

- Error (5 Kategori):
  - Very Small = < -40
  - Small = -50 - -25
  - Medium = -30 – 30
  - Big = 20 – 40
  - Very Big = >40
- Del Error (5 Kategori):
  - Very Small = < -40
  - Small = -30 - -15
  - Medium = -20 – 20
  - Big = 15 – 30
  - Very Big = >40

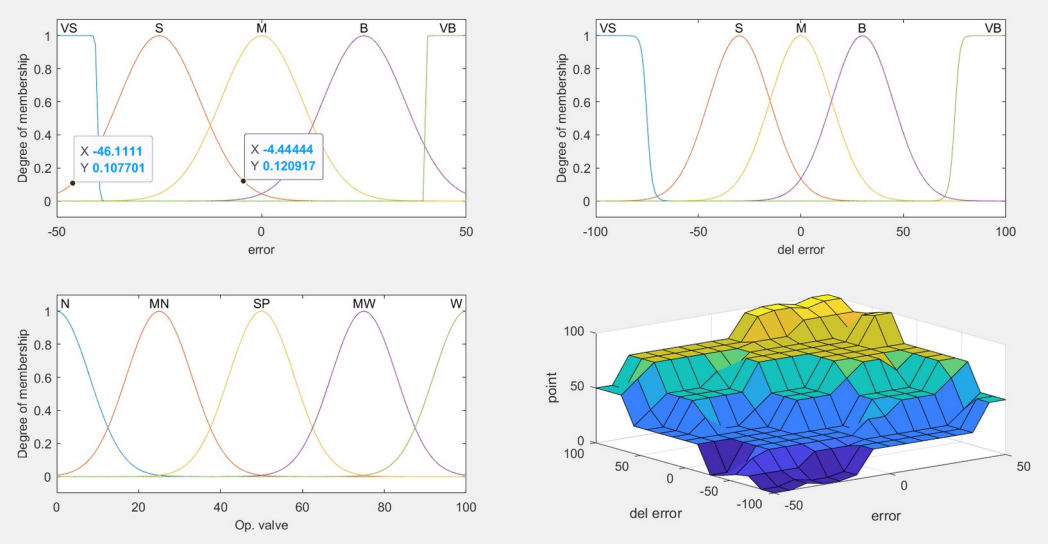
Output:

- Percent Opening Valve (5 Kategori):
  - Narrow = < 25
  - Medium Narrow = 20 – 50
  - Set Point = 30 – 60
  - Medium Wide = 50 – 80
  - Wide = >75

c. Rule Base

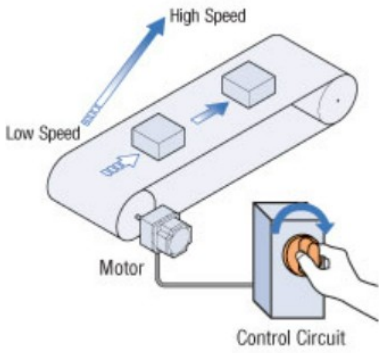
Rule based/aturan dari FIS dapat disusun dalam table sebagai berikut:

e\ de	VS	S	M	B	VB
VS	N	N	MN	MN	SP
S	N	MN	MN	SP	MW
M	MN	MN	SP	MW	MW
B	MN	SP	MW	MW	W
VB	SP	MW	MW	W	W



3. Kontrol Conveyor Belt pada plant Penyeleksi barang
   a. Deskripsi

      Pada conveyor belt penyeleksi barang dengan beban maksimum 150 Kg memiliki kemampuan untuk menyesuaikan kecepatannya. Tujuannya adalah efisiensi waktu dan kesesuaian output. Jika beban ringan, maka speed conveyor dikisaran pelan agar barang tidak terlempar, ketika beban medium maka diset medium juga speednya agar efisien waktu, dan ketika beban berat maka kecepatannya dipercepat agar barang cepat sampai di container penyeleksi.



b. Membership Function

Input (1 parameter):

- Temperature Furnace (3 Kategori):
  - light = 0 – 40 Kg
  - medium = 20 – 110 Kg
  - heavy= >100 Kg

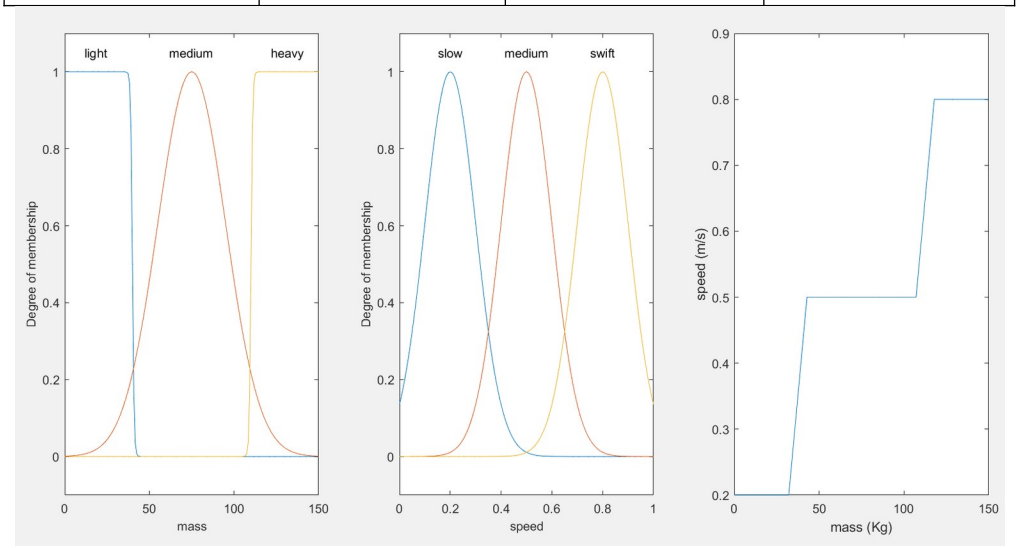
Output (1 parameter):

- Crude Oil Flow (3 Kategori):
  - slow = 0 – 0.4 m/s
  - medium = 0.3 – 0.8 m/s
  - swift = 0.7 – 1 m/s

c. Rule Base

Rule based/aturan dari FIS dapat disusun dalam table sebagai berikut:

Input:	Light	Medium	Heavy
Output:	Slow	Medium	Swift



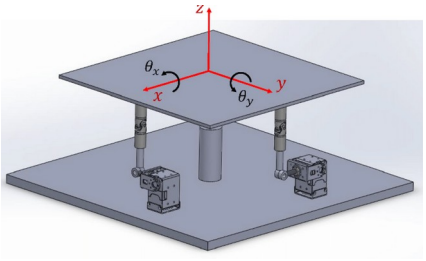
4. Kontrol Sudut Sistem Ball and Plate

a. Deskripsi

Sistem ball and plate merupakan realisasi dari model kesetimbangan fisik dari dinamika benda tegar. Tujuan dari

sistem ball and plate adalah menjaga posisi bola berada pada tengah plate 2D sehingga bola tidak jatuh ke pinggir plate. Oleh karenanya diperlukan pengaturan sudut kemiringan beam dalam menyesuaikan posisi ball terhadap plate.

Input dari sistem adalah posisi pola terhadap plate, sedangkan outputnya adalah besar sudut actuator yang diberikan plate.



b. Membership Function

Input (2 parameter):

- Position x (3 Kategori):
  - Negative x = -3 – 0 cm
  - Stable x = -2 – 2 cm
  - Positive x= 0 – 3 cm
- Position y (3 Kategori):
  - Negative y = -3 – 0 cm
  - Stable y = -2 – 2 cm
  - Positive y= 0 – 3 cm

Output (2 parameter):

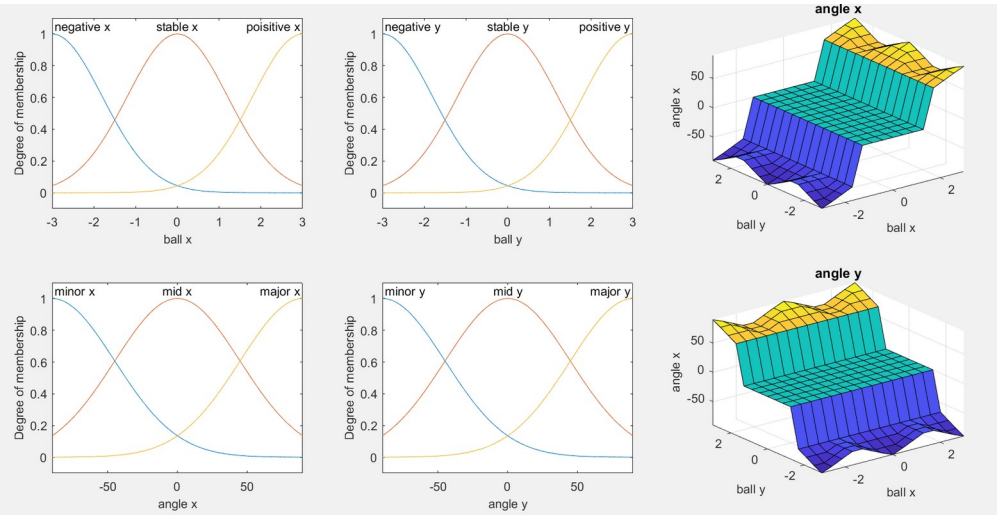
- Angle x (3 Kategori):
  - slow = -90 – -45 m/s
  - medium = -50 – 50 m/s
  - swift = 45 – 90 m/s
- Angle x (3 Kategori):
  - slow = -90 – -45 m/s
  - medium = -50 – 50 m/s
  - swift = 45 – 90 m/s

c. Rule Base

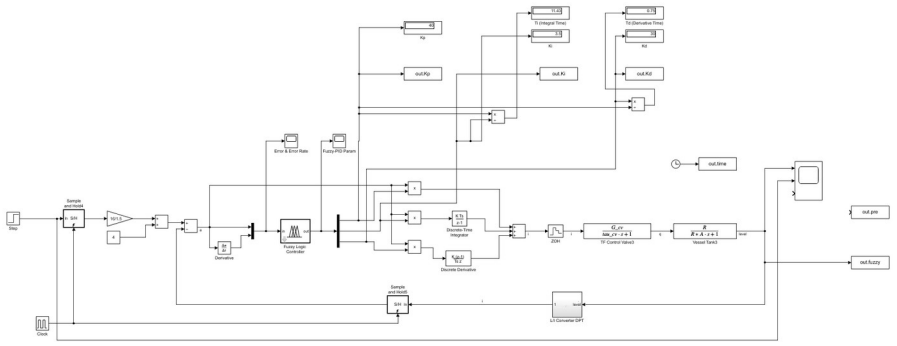
Rule based/aturan dari FIS dapat disusun dalam table sebagai berikut:

	plate x	-3	0	3
	ball x\ y	ny	s	py
-3	nx	minor x	minor x	minor x
0	s	mid	mid	mid
3	px	major x	major x	major x

	plate y	-3	0	3
	ball x\ y	ny	s	py
-3	nx	minor y	mid	major y
0	s	minor y	mid	major y
3	px	minor y	mid	major y



5. Adjustment Parameter PID pada Kontrol Level Tank
   a. Deskripsi
      Pada sistem AFPIDC, controller dapat menyesuaikan nilai paraSmeter PID terhadap perubahan yang terjadi di plant, baik perubahan parameter maupun disturbance. Oleh karenanya fuzzy inference system mengatur nilai keluaran Kp, Ki, Kd dengan input dari error dan delta error dari sistem orde 2 pada kendali level tank.



   b. Membership Function
      Input (2 Parameter):
      - Error (5 Kategori):
        - Negative Big = < -0.5

    - Negative Small = -0.7 - -0.2
    - Zero = -0.3 – 0.3
    - Positive Small = 0.2 – 0.7
    - Positive Big = >0.5
- Del Error (5 Kategori):
    - Negative Big = < -5000
    - Negative Small = -6000 - -2000
    - Zero = -3000 – 3000
    - Positive Small = 2000 – 7000
    - Positive Big = >5000

Output:

- Paramter Kp (5 Kategori):
    - Negative Big = 0 – 20
    - Negative Small = 15 – 35
    - Zero = 30 – 55
    - Positive Small = 50 – 65
    - Positive Big = >60
- Paramter Ki (5 Kategori):
    - Negative Big = 0 – 1
    - Negative Small = 1 – 3
    - Zero = 2.5 – 4.5
    - Positive Small = 4 – 6
    - Positive Big = 5 – 7
- Paramter Kd (5 Kategori):
    - Negative Big = 0 – 20
    - Negative Small = 15 – 30
    - Zero = 25 – 35
    - Positive Small = 30 – 55
    - Positive Big = 50 – 60

c. Rule Base

Rule based/aturan dari FIS dapat disusun dalam table sebagai berikut:

Rule Base Parameter Kp

E|DE	***NB***	***NS***	***ZE***	***PS***	***PB***
NB	NB	NB	NS	NB	NB
NS	NB	NB	NS	NB	NB
ZE	NS	NS	ZE	NS	NS
PS	PB	PB	PS	PB	PB
PB	PB	PB	PS	PB	PB

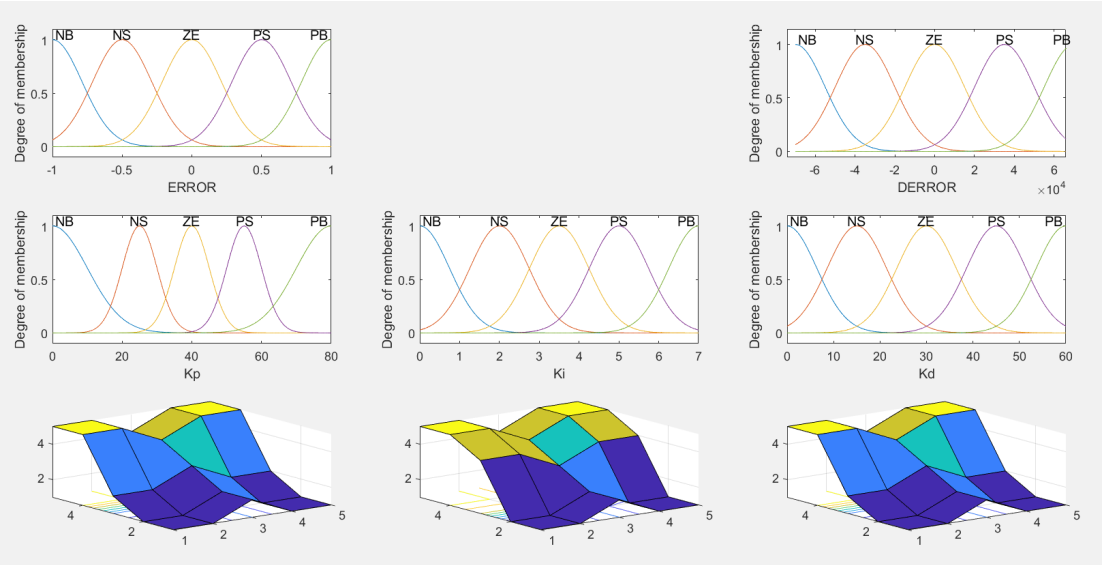
Rule Base Parameter Ki

E|DE	***NB***	***NS***	***ZE***	***PS***	***PB***

NB	NB	NB	NS	NB	NB
NS	NB	NB	NS	NB	NB
ZE	PS	PS	ZE	PS	PS
PS	PB	PB	PS	PB	PB
PB	PB	PB	PS	PB	PB

Rule Base Parameter Kd

E|DE	***NB***	***NS***	***ZE***	***PS***	***PB***
NB	NB	NB	NS	NB	NB
NS	NB	NB	NS	NB	NB
ZE	NS	NS	ZE	NS	NS
PS	PB	PB	PS	PB	PB
PB	PB	PB	PS	PB	PB



Coding MATLAB

```
%% Fuzzy Logic: Opening Control Valve
clear; clc;

%Set point: 50% Opening (Medium Opening)

%Set Up Fuzzy
fuzzyvalve = mamfis('Name',"FuzzyValve");
fuzzyvalve.DefuzzificationMethod="mom";
fuzzyvalve = addInput(fuzzyvalve,[-50 50],'Name',"error");
%Error
fuzzyvalve = addInput(fuzzyvalve,[-100 100],'Name',"del error");
%Different error
fuzzyvalve = addOutput(fuzzyvalve,[0 100],'Name',"Op. valve");
%Opening Valve

%membership function of masukan 1, gaussmf: [std.dev mean]
sde = 10;
```

```
fuzzyvalve = addMF(fuzzyvalve,"error","gbellmf",[sde 25 -
50],'Name',"VS");      %Very Small = 1
fuzzyvalve = addMF(fuzzyvalve,"error","gaussmf",[sde -
25],'Name',"S");         %Small = 2
fuzzyvalve = addMF(fuzzyvalve,"error","gaussmf",[sde 0],'Name',"M");
%Medium = 3
fuzzyvalve = addMF(fuzzyvalve,"error","gaussmf",[sde
25],'Name',"B");         %Big = 4
fuzzyvalve = addMF(fuzzyvalve,"error","gbellmf",[sde 50
50],'Name',"VB");    %Very Big = 5

%membership function of masukan 2
sdde= 25;
fuzzyvalve = addMF(fuzzyvalve,"del error","gbellmf",[sdde 10 -
100],'Name',"VS");   %Very Small = 1
fuzzyvalve = addMF(fuzzyvalve,"del error","gaussmf",[15 -
30],'Name',"S");           %Small = 2
fuzzyvalve = addMF(fuzzyvalve,"del error","gaussmf",[15
0],'Name',"M");              %Medium = 3
fuzzyvalve = addMF(fuzzyvalve,"del error","gaussmf",[15
30],'Name',"B");            %Big = 4
fuzzyvalve = addMF(fuzzyvalve,"del error","gbellmf",[sdde 10
100],'Name',"VB");    %Very Big = 5

%membership function of keluaran
sdo = 8;
fuzzyvalve = addMF(fuzzyvalve,"Op. valve","gaussmf",[sdo
0],'Name',"N");         %Narrow = 1
fuzzyvalve = addMF(fuzzyvalve,"Op. valve","gaussmf",[sdo
25],'Name',"MN");       %Medium Narrow = 2
fuzzyvalve = addMF(fuzzyvalve,"Op. valve","gaussmf",[sdo
50],'Name',"SP");       %Set Point = 3
fuzzyvalve = addMF(fuzzyvalve,"Op. valve","gaussmf",[sdo
75],'Name',"MW");       %Medium Wide = 4
fuzzyvalve = addMF(fuzzyvalve,"Op. valve","gaussmf",[sdo
100],'Name',"W");       %Wide = 5

%
rules_opvalve =[1 1 1 1 1;
1 2 1 1 1;
1 3 2 1 1;
1 4 2 1 1;
1 5 3 1 1;
2 1 1 1 1;
2 2 2 1 1;
2 3 2 1 1;
2 4 3 1 1;
2 5 4 1 1;
3 1 2 1 1;
3 2 2 1 1;
3 3 3 1 1;
3 4 4 1 1;
3 5 4 1 1;
4 1 2 1 1;
```

```
4 2 3 1 1;
4 3 4 1 1;
4 4 4 1 1;
4 5 5 1 1;
5 1 3 1 1;
5 2 4 1 1;
5 3 4 1 1;
5 4 5 1 1;
5 5 5 1 1];

rules = rules_opvalve;

value_opvalve = rules_opvalve(:,3)';
value_opvalve_surf=[value_opvalve(1:5);
    value_opvalve(6:10);
    value_opvalve(11:15);
    value_opvalve(16:20);
    value_opvalve(21:25)];
[e,de] = meshgrid(1:5,1:5);

%adding rules to the fis
fuzzyvalve = addRule(fuzzyvalve,rules);

figure(1)
subplot(2,2,1)
plotmf(fuzzyvalve,'input',1)
subplot(2,2,2)
plotmf(fuzzyvalve,'input',2)
subplot(2,2,3)
plotmf(fuzzyvalve,'output',1)
subplot(2,2,4)
gensurf(fuzzyvalve)
%surfc(e,de,value_opvalve_surf)
xlabel('error')
ylabel('del error')
zlabel('point')

%% Fuzzy Logic: Furnance Temperature Control
%Set point: 250 degC (Optimal Burning Temperature)

%Set Up Fuzzy
fuzzyfurnance = mamfis('Name',"fuzzyfurnance");
fuzzyfurnance.DefuzzificationMethod="mom";
fuzzyfurnance = addInput(fuzzyfurnance,[0 1500],'Name',"temp");
%temp
fuzzyfurnance = addInput(fuzzyfurnance,[-10 10],'Name',"expansion");
%Different temp
fuzzyfurnance = addOutput(fuzzyfurnance,[0 100],'Name',"flow gas");
%Opening Valve

%membership function of input temperature (in degC)
sde = 150;
fuzzyfurnance = addMF(fuzzyfurnance,"temp","gaussmf",[sde
0],'Name',"Warm");        %1
```

```
fuzzyfurnance = addMF(fuzzyfurnance,"temp","gaussmf",[sde
750],'Name',"Hot");       %2
fuzzyfurnance = addMF(fuzzyfurnance,"temp","gaussmf",[sde
1500],'Name',"Too Hot");  %3

%membership function of input expansionion (in cm)
sdde= 3;
fuzzyfurnance = addMF(fuzzyfurnance,"expansion","gaussmf",[2 -
10],'Name',"under exp");    %1
fuzzyfurnance = addMF(fuzzyfurnance,"expansion","gaussmf",[3
0],'Name',"normal");         %2
fuzzyfurnance = addMF(fuzzyfurnance,"expansion","gaussmf",[2
10],'Name',"over exp");      %3

%membership function of flow gas
sdo = 15;
fuzzyfurnance = addMF(fuzzyfurnance,"flow gas","gaussmf",[sdo
0],'Name',"low");       %1
fuzzyfurnance = addMF(fuzzyfurnance,"flow gas","gaussmf",[sdo
50],'Name',"medium");   %2
fuzzyfurnance = addMF(fuzzyfurnance,"flow gas","gaussmf",[sdo
100],'Name',"fast");    %3


rules_furnancetemp =[1 1 3 1 1;
1 2 3 1 1;
1 3 2 1 1;
2 1 3 1 1;
2 2 2 1 1;
2 3 2 1 1;
3 1 2 1 1;
3 2 1 1 1;
3 3 1 1 1];

rules = rules_furnancetemp;

%adding rules to the fis
fuzzyfurnance = addRule(fuzzyfurnance,rules);

figure(2)
subplot(2,2,1)
plotmf(fuzzyfurnance,'input',1)
subplot(2,2,2)
plotmf(fuzzyfurnance,'input',2)
subplot(2,2,3)
plotmf(fuzzyfurnance,'output',1)
subplot(2,2,4)
gensurf(fuzzyfurnance)
%surfc(e,de,value_furnancetemp_surf)
xlabel('temp (degC)')
ylabel('expansion (mm)')
zlabel('magnitude')

%% Fuzzy Logic: Conveyor Belt
```

```
%Set Up Fuzzy
fuzzyconveyor = mamfis('Name',"fuzzyconveyor");
fuzzyconveyor.DefuzzificationMethod="mom";
fuzzyconveyor = addInput(fuzzyconveyor,[0 150],'Name',"mass");
%mass
fuzzyconveyor = addOutput(fuzzyconveyor,[0 1],'Name',"speed");
%conveyor speed

%membership function of input mass (in Kg)
sde = 20;
fuzzyconveyor = addMF(fuzzyconveyor,"mass","gbellmf",[40 40
0],'Name',"light");       %1
fuzzyconveyor = addMF(fuzzyconveyor,"mass","gaussmf",[sde
75],'Name',"medium");       %2
fuzzyconveyor = addMF(fuzzyconveyor,"mass","gbellmf",[40 40
150],'Name',"heavy");     %3

%membership function of output conveyor speed (in m/s)
sdo = 0.1;
fuzzyconveyor = addMF(fuzzyconveyor,"speed","gaussmf",[sdo
0.2],'Name',"slow");       %1
fuzzyconveyor = addMF(fuzzyconveyor,"speed","gaussmf",[sdo
0.5],'Name',"medium");     %2
fuzzyconveyor = addMF(fuzzyconveyor,"speed","gaussmf",[sdo
0.8],'Name',"swift");      %3


rules_conveyorbelt =[1 1 1 1;
2 2 1 1;
3 3 1 1];

rules = rules_conveyorbelt;

%adding rules to the fis
fuzzyconveyor = addRule(fuzzyconveyor,rules);

figure(3)
subplot(1,3,1)
plotmf(fuzzyconveyor,'input',1)
subplot(1,3,2)
plotmf(fuzzyconveyor,'output',1)
subplot(1,3,3)
gensurf(fuzzyconveyor)
%surfc(e,de,value_furnancemass_surf)
xlabel('mass (Kg)')
ylabel('speed (m/s)')

%% Fuzzy Logic: Ball and Plate
%Set point: posisi (x,y) = (0,0)

%Set Up Fuzzy
fuzzy_ballplate = mamfis('Name',"fuzzy_ballplate");
fuzzy_ballplate.DefuzzificationMethod="mom";
```

```
fuzzy_ballplate = addInput(fuzzy_ballplate,[-3 3],'Name',"ball x");
fuzzy_ballplate = addInput(fuzzy_ballplate,[-3 3],'Name',"ball y");
fuzzy_ballplate = addOutput(fuzzy_ballplate,[-90 90],'Name',"angle
x");
fuzzy_ballplate = addOutput(fuzzy_ballplate,[-90 90],'Name',"angle
y");

%membership function of position ball x
sde = 1.2;
fuzzy_ballplate = addMF(fuzzy_ballplate,"ball x","gaussmf",[sde -
3],'Name',"negative x");     %1
fuzzy_ballplate = addMF(fuzzy_ballplate,"ball x","gaussmf",[sde
0],'Name',"stable x");        %2
fuzzy_ballplate = addMF(fuzzy_ballplate,"ball x","gaussmf",[sde
3],'Name',"poisitive x");     %3

%membership function of position ball y
sdde= 1.2;
fuzzy_ballplate = addMF(fuzzy_ballplate,"ball y","gaussmf",[sdde -
3],'Name',"negative y");           %1
fuzzy_ballplate = addMF(fuzzy_ballplate,"ball y","gaussmf",[sdde
0],'Name',"stable y");              %2
fuzzy_ballplate = addMF(fuzzy_ballplate,"ball y","gaussmf",[sdde
3],'Name',"positive y");            %3

%membership function of output angle x
sdo = 45;
fuzzy_ballplate = addMF(fuzzy_ballplate,"angle x","gaussmf",[sdo -
90],'Name',"minor x");   %1
fuzzy_ballplate = addMF(fuzzy_ballplate,"angle x","gaussmf",[sdo
0],'Name',"mid x");       %2
fuzzy_ballplate = addMF(fuzzy_ballplate,"angle x","gaussmf",[sdo
90],'Name',"major x");    %3

%membership function of output angle y
sdo = 45;
fuzzy_ballplate = addMF(fuzzy_ballplate,"angle y","gaussmf",[sdo -
90],'Name',"minor y");   %1
fuzzy_ballplate = addMF(fuzzy_ballplate,"angle y","gaussmf",[sdo
0],'Name',"mid y");       %2
fuzzy_ballplate = addMF(fuzzy_ballplate,"angle y","gaussmf",[sdo
90],'Name',"major y");    %3


rules_ballplate =[1 1 1 1 1 1;
1 2 1 2 1 1;
1 3 1 3 1 1;
2 1 2 1 1 1;
2 2 2 2 1 1;
2 3 2 3 1 1;
3 1 3 1 1 1;
3 2 3 2 1 1;
3 3 3 3 1 1];
```

```
rules_ballplate_x = rules_ballplate;
rules_ballplate_y = [rules_ballplate(:,1) rules_ballplate(:,2)
rules_ballplate(:,4) rules_ballplate(:,5) rules_ballplate(:,6)]

%adding rules to the fis
fuzzy_ballplatex = addRule(fuzzy_ballplate,rules_ballplate_x);
fuzzy_ballplatey = addRule(fuzzy_ballplate,rules_ballplate_y);

figure(4)
subplot(2,3,1)
plotmf(fuzzy_ballplate,'input',1)
subplot(2,3,2)
plotmf(fuzzy_ballplate,'input',2)
subplot(2,3,3)
gensurf(fuzzy_ballplatex)
title('angle x')
subplot(2,3,4)
plotmf(fuzzy_ballplate,'output',1)
subplot(2,3,5)
plotmf(fuzzy_ballplate,'output',2)
subplot(2,3,6)
gensurf(fuzzy_ballplatey)
title('angle y')

%% Fuzzy Logic: AFPIDC Level Control
clear; clc;

%Set point: 0.75m

fuzzypid = mamfis('Name',"AFPIDC");
fuzzypid.DefuzzificationMethod="mom";
fuzzypid = addInput(fuzzypid,[-1 1],'Name',"ERROR"); %Adding input 1
fuzzypid = addInput(fuzzypid, [-7e+4 7e+4],'Name',"DERROR"); %Adding
input 2
fuzzypid = addOutput(fuzzypid,[0 80],'Name',"Kp"); %Kp
fuzzypid = addOutput(fuzzypid, [0 7],'Name',"Ki"); %Ki
fuzzypid = addOutput(fuzzypid,[0 60],'Name',"Kd"); %Kd

%membership function of input error
fuzzypid = addMF(fuzzypid,"ERROR","gaussmf",[0.2122 -
1],'Name',"NB"); %Negative Big = 1
fuzzypid = addMF(fuzzypid,"ERROR","gaussmf",[0.2122 -
0.5],'Name',"NS"); %Negative Small =2
fuzzypid = addMF(fuzzypid,"ERROR","gaussmf", [0.2122
0],'Name',"ZE"); %Zero =3
fuzzypid = addMF(fuzzypid,"ERROR","gaussmf", [0.2122
0.5],'Name',"PS"); %Positive Small =4
fuzzypid = addMF(fuzzypid,"ERROR","gaussmf", [0.2122
1],'Name',"PB"); %Positive Big =5

%membership function of input derror
fuzzypid = addMF(fuzzypid,"DERROR","gaussmf", [1.487e+4 -
7e+4],'Name',"NB"); %Negative Big=1
```

```
fuzzypid = addMF(fuzzypid,"DERROR","gaussmf", [1.487e+4 -
3.5e+4],'Name',"NS"); %Negative Small=2
fuzzypid = addMF(fuzzypid,"DERROR","gaussmf", [1.487e+4
0],'Name',"ZE"); %Zero=3
fuzzypid = addMF(fuzzypid,"DERROR","gaussmf", [1.487e+4
3.5e+4],'Name',"PS"); %Positive Small=4
fuzzypid = addMF(fuzzypid,"DERROR","gaussmf", [1.487e+4
7e+4],'Name',"PB"); %Positive Big=5

%membership function of output KP
fuzzypid = addMF(fuzzypid,"Kp","gaussmf", [10 0],'Name',"NB");
%Negative Big=1
fuzzypid = addMF(fuzzypid,"Kp","gaussmf", [5 25],'Name',"NS");
%Negative Small=2
fuzzypid = addMF(fuzzypid,"Kp","gaussmf", [5 40],'Name',"ZE");
%Zero=3
fuzzypid = addMF(fuzzypid,"Kp","gaussmf", [5 55],'Name',"PS");
%Positive Small=4
fuzzypid = addMF(fuzzypid,"Kp","gaussmf", [10 80],'Name',"PB");
%Positive Big=5

%membership function of output KI
fuzzypid = addMF(fuzzypid,"Ki","gaussmf", [0.7434 0],'Name',"NB");
%Negative Big =1
fuzzypid = addMF(fuzzypid,"Ki","gaussmf", [0.742 2],'Name',"NS");
%Negative Small=2
fuzzypid = addMF(fuzzypid,"Ki","gaussmf", [0.742 3.5],'Name',"ZE");
%Zero=3
fuzzypid = addMF(fuzzypid,"Ki","gaussmf", [0.742 5],'Name',"PS");
%Positive Small=4
fuzzypid = addMF(fuzzypid,"Ki","gaussmf", [0.7434 7],'Name',"PB");
%Positive Big=5

%membership function of output KD
fuzzypid = addMF(fuzzypid,"Kd","gaussmf", [6.37 0],'Name',"NB");
%Negative Big=1
fuzzypid = addMF(fuzzypid,"Kd","gaussmf", [6.37 15],'Name',"NS");
%Negative Small=2
fuzzypid = addMF(fuzzypid,"Kd","gaussmf", [6.37 30],'Name',"ZE");
%Zero=3
fuzzypid = addMF(fuzzypid,"Kd","gaussmf", [6.37 45],'Name',"PS");
%Positive Small=4
fuzzypid = addMF(fuzzypid,"Kd","gaussmf", [6.37 60],'Name',"PB");
%Positive Big=5

rules_kp =[1 1 1 0 0 1 1;
1 2 1 0 0 1 1;
1 3 2 0 0 1 1;
1 4 1 0 0 1 1;
1 5 1 0 0 1 1;
2 1 1 0 0 1 1;
2 2 1 0 0 1 1;
2 3 2 0 0 1 1;
2 4 1 0 0 1 1;
```

```
2 5 1 0 0 1 1;
3 1 2 0 0 1 1;
3 2 2 0 0 1 1;
3 3 3 0 0 1 1;
3 4 2 0 0 1 1;
3 5 2 0 0 1 1;
4 1 5 0 0 1 1;
4 2 5 0 0 1 1;
4 3 4 0 0 1 1;
4 4 5 0 0 1 1;
4 5 5 0 0 1 1;
5 1 5 0 0 1 1;
5 2 5 0 0 1 1;
5 3 4 0 0 1 1;
5 4 5 0 0 1 1;
5 5 5 0 0 1 1];

rules_ki_pre=[1 1 1 0 0 1 1;
1 2 1 0 0 1 1;
1 3 2 0 0 1 1;
1 4 1 0 0 1 1;
1 5 1 0 0 1 1;
2 1 1 0 0 1 1;
2 2 1 0 0 1 1;
2 3 2 0 0 1 1;
2 4 1 0 0 1 1;
2 5 1 0 0 1 1;
3 1 4 0 0 1 1;
3 2 4 0 0 1 1;
3 3 3 0 0 1 1;
3 4 4 0 0 1 1;
3 5 4 0 0 1 1;
4 1 5 0 0 1 1;
4 2 5 0 0 1 1;
4 3 4 0 0 1 1;
4 4 5 0 0 1 1;
4 5 5 0 0 1 1;
5 1 5 0 0 1 1;
5 2 5 0 0 1 1;
5 3 4 0 0 1 1;
5 4 5 0 0 1 1;
5 5 5 0 0 1 1];

value_kp = rules_kp(:,3)';
value_kp_surf=[value_kp(1:5);
    value_kp(6:10);
    value_kp(11:15);
    value_kp(16:20);
    value_kp(21:25)];

n_ki = [1 2 4 3 5 6 7];
rules_ki = rules_ki_pre(:,n_ki);
value_ki = rules_ki(:,4)';
value_ki_surf=[value_ki(1:5);
```

```
    value_ki(6:10);
    value_ki(11:15);
    value_ki(16:20);
    value_ki(21:25)];

n_kd = [1 2 5 4 3 6 7];
rules_kd = rules_kp(:,n_kd);
value_kd = rules_kd(:,5)';
value_kd_surf=[value_kp(1:5);
    value_kd(6:10);
    value_kd(11:15);
    value_kd(16:20);
    value_kd(21:25)];

[e,de] = meshgrid(1:5,1:5);

rules = [rules_kp;rules_ki;rules_kd];

%adding rules to the fis
fuzzypid = addRule(fuzzypid,rules);

figure(5)
subplot(3,3,1)
plotmf(fuzzypid,'input',1)
subplot(3,3,3)
plotmf(fuzzypid,'input',2)
subplot(3,3,4)
plotmf(fuzzypid,'output',1)
subplot(3,3,5)
plotmf(fuzzypid,'output',2)
subplot(3,3,6)
plotmf(fuzzypid,'output',3)
subplot(3,3,7)
surfc(e,de,value_kp_surf)
subplot(3,3,8)
surfc(e,de,value_ki_surf)
subplot(3,3,9)
surfc(e,de,value_kd_surf)
```